cc-68-33

✶SORTER - A Simulated Card Sorter for OS-3

by Computer Center Staff

September, 1968

OSU
COMPUTER CENTER
Oregon State University
Corvallis, Oregon 97331

*SORTER - A Simulated Card Sorter for OS-3

cc-68-33

by

Computer Center Staff

September, 1968

Computer Center
Oregon State University
Corvallis, Oreg. 97331

*SORTER

*SORTER is a program that simulates the operation of a card sorter. It runs under the OS-3 operating system on the CDC 3300.

Input to the *SORTER program is a file of variable length BCD records (80 column keypunch cards, for example); the maximum length of any record is 136 characters. An input file may be read from the card reader or it may be stored in the system as a saved file. In the latter case, the saved file may be prepared as a card deck and copied into a saved file, or the on-line editor can be used to create the file. For a definition of saved files and instructions on the use of EDIT, the user is referred to:

1) OS-3 User's Manual, cc-68-3

2) A Control Mode Manual for OS-3 Version 2.0, cc-68-21

3) OS-3 Teletype Editor Manual, cc-68-17

The format of a program for saving a card deck under some file name is given here:*

```
7<Job #>,<User #>,<Identification field>
8

7 Copy, Ø=<FILE NAME>
8
                 }
            card deck here
                 }
77
88

7 LOGOFF
8
```

* The characters "<" and ">" in the above prototype are used to denote fields; these characters should not be punched in the card. For example: John Doe has a job no. = 99999 and a user no. = 1234. His first card in the preceeding prototype looks like this:

```
7
8 99999,1234,John Doe
```

Prior to using *SORTER, the input file must be equipped to a logical unit; the program rewinds the input file if necessary. Program output is stored in 13 logical units that correspond to the 13 pockets of a sorter. Pockets and their logical unit equivalents are given below:

Pocket	LUN
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
11	11
12	12
Reject	10

Upon completion of the program, a file mark is written at the end of each output file and the file is rewound.

The calling sequence for *SORTER is:

```
*SORTER,I=<lun>,T=<type>,C=<column>,P=<lun>
```

where I = the input logical unit. If I is given, but no unit is specified, then logical unit 60 is assumed.

T = A number between 1 and 5 that designates the type of sort to be performed.

1. [0-9] punch → Pockets 0 through 9

   Blank → Reject

   All else → Reject

If there is more than one digit punch in the column, sorting takes place on the smallest digit punched. All zone punches ignored except 0 if no other punches in the column.

2. [0-9] punch → Reject

   12 punch → Pocket 12

   11 punch → Pocket 11

   0 punch → Pocket 0

   All else → Reject

3. [0-9] punch → Reject

   [A-I] punch → Pockets 1 through 9

   [J-R] punch → Pocket 11

   [S-Z] punch → Pocket 0

   All else → Reject

4. [0-9] punch → Reject

   [J-R] punch → Pockets 1 through 9

   All else → Reject

5. [0-9] punch → Pockets 0 through 9

   [A-I] punch → Pocket 12

   [J-R] punch → Pocket 11

   [S-Z] punch → Reject

   All else → Reject

C = The number of the column upon which the sort will be performed. If no column is named, column 1 is assumed.

P = The logical unit on which totals will be written. If no unit is specified, the standard output unit = 61 will be assumed. If P is not present, totals will be suppressed.

If further sorting is required on one of the output logical units, *SORTER can be called again with this unit as input. Before proceeding, the program will automatically transfer any input logical unit in the range 0-12 to another logical unit and destroy residual information contained in 0-12.

At the conclusion of the program, EDIT may be used to inspect the contents of any output files. By use of COPY, these files may also be listed on the printer or punched as a new deck.

EXAMPLES:

1. This program sorts on columns 1-3 numerically. Data is entered on the teletypewriter.

2. This program sorts on column 1-3 alphabetically. Data file is entered on the teletypewriter.

3. This program sorts on columns 1-3 alphabetically. Data file is entered through the card reader.

Turn Teletype to the on-line position. Press the keys control and A at the same time. A # will be generated. Then type in your job number and user number and press carriage return (CR). Your number will be blotted out for security reasons and another # will be printed. Then type in EDIT CR . "]" will be printed and you will type INPUT CR . Now you are ready to type in the records to be sorted. Each line in the example represents one record. Records are sequentially numbered by the EDIT program.

EXAMPLE 1

```
[redacted]                                      Explanation
#EDIT

]INPUT
ØØØØ1:579
ØØØØ2:39Ø
ØØØØ3:491
ØØØØ4:169
ØØØØ5:295
ØØØØ6:395
ØØØØ7:296
ØØØØ8:834
ØØØØ9:9ØØ
ØØØ1Ø:935                                  ENTRY OF DATA
ØØØ11:517
ØØØ12:322
ØØØ13:911
ØØØ14:Ø67
ØØØ15:283
ØØØ16:89Ø
ØØØ17:525
ØØØ18:821
ØØØ19:723
ØØØ2Ø:ØØØ
ØØØ21:
]OUT,DATA  (Press cntrl W or alt mode to get back into EDIT mode)

]EQUIP,33=DATA

]          (Press cntrl A to get back to control mode)

#*SORTER,I=33,C=3,T=1,P    (This initiates sort on column 3)
COUNTER VALUES-
      Ø=          4
      1=          3
      2=          1
      3=          2
      4=          1
      5=          4
      6=          1                    (SORTING ON COLUMN 3)
      7=          2
      8=          Ø
      9=          2
     11=          Ø
     12=          Ø
   REJECT=        Ø
   TOTAL=        2Ø

#
```

```
#EDIT
]FIN,Ø
]APPEND,1
]APPEND,2
]APPEND,3
]APPEND,4
]APPEND,5
]APPEND,6
]APPEND,7
]APPEND,8
]APPEND,9
]OUT,TEST
```

(File input: scratch area (this file contains the records with zeroes in column 3). Add to this, file 1 which contains the ones in column 3, by appending file 1, etc. ... through file 9. Then the completed file is written by the output command under the name of Test which is also a file.)

```
]EQUIP,34=TEST
]
```

(In order to sort in column 2, we must equip the file named Test = to a logical unit number n, where 12<n<50.

(Press cntrl A and proceed to sort column 2)

```
#*SORTER,I=34,C=2,T=1
```

(This initiates sorting on column 2; since P was not present in *SORTER statement, the totals (counter values) will not be printed. The program proceeds exactly as the sort on col 3.)

```
#EDIT

]FIN,Ø

]APPEND,1

]APPEND,2

]APPEND,3

]APPEND,4

]APPEND,5

]APPEND,6

]APPEND,7

]APPEND,8

]APPEND,9

]OUT,TEST

]
#*SORTER,I=34,C=1,T=1,P
COUNTER VALUES-
        Ø=            2
        1=            1
        2=            3
        3=            3
        4=            1
        5=            3
        6=            0
        7=            1
        8=            3
        9=            3
       11=            Ø
       12=            Ø
    REJECT=           Ø
    TOTAL=           2Ø
```

(SORTING ON COLUMN 1)

```
#EDIT

]
]FIN,Ø

]APPEND,1

]APPEND,2

]APPEND,3

]APPEND,4

]APPEND,5

]APPEND,6

]APPEND,7

]APPEND,8

]APPEND,9

]OUT,TEST

]LIST                (The final listing shows numbers completely
        :ØØØ          sorted from columns 1-3.)
        :Ø67
        :169
        :283
        :295
        :296
        :322
        :39Ø
        :395
        :491
        :517
        :525
        :579
        :723
        :821
        :834
        :89Ø
        :9ØØ
        :911
        :935

]
#LOGOFF
TIME 4.5ØØ SECONDS  NFBLKS 25  COST $Ø.52
#
```

EXAMPLE 2

```
#EDIT

]INPUT
00001:QWE                        (DATA ENTERED ON TELETYPEWRITER)
00002:WER
00003:ERT
00004:RTY
00005:TYU
00006:YUI
00007:UIO
00008:IOP
00009:ASD
00010:SDF
00011:FGH
00012:GHJ
00013:HJK
00014:JKL
00015:ZXC
00016:XCV
00017:CVB
00018:BNM
00019:NMQ
00020:MQW
00021:DFG
00022:VBN
00023:
]OUT,TEST

]
#EQUIP,34=TEST
#*SORTER,I=34,C=3,T=1,P          (SORTING ON COLUMN 3 DONE NUMERICALLY
COUNTER VALUES-                   FIRST.)
       0=          0
       1=          1
       2=          2
       3=          3
       4=          3
       5=          3
       6=          3
       7=          2
       8=          3
       9=          2
      11=          0
      12=          0
   REJECT=         0
   TOTAL=         22
```

```
#EDIT

]FIN 0

]APPEND,1

]APPEND,2

]APPEND,3

]APPEND,4

]APPEND,5

]APPEND,6

]APPEND,7

]APPEND,8

]APPEND,9

]OUT,TEST

]


#*SORTER,I=34,C=3,T=2,P         (SORTING DONE ON ZONES IN COLUMN 3.)
COUNTER VALUES-
        0=            5
        1=            0
        2=            0
        3=            0
        4=            0
        5=            0
        6=            0
        7=            0
        8=            0
        9=            0
       11=            9
       12=            8
    REJECT=           0
    TOTAL=           22

#EDIT

]FIN 12

]APPEND,11

]APPEND,0

]OUT,TEST                       (NOW COLUMN 3 IS SORTED ALPHABETICALLY.)

]
```

```
#*SORTER,I=34,C=2,T=1,P          (SAME PROCEDURE FOR COLUMN 2.)
COUNTER VALUES-
      0=          0
      1=          1
      2=          3
      3=          2
      4=          3
      5=          3
      6=          3
      7=          2
      8=          3
      9=          2
     11=          0
     12=          0
   REJECT=        0
   TOTAL=        22

#EDIT

]FIN 0

]APPEND,1

]APPEND,2

]APPEND,3

]APPEND,4

]APPEND,5

]APPEND,6

]APPEND,7

]APPEND,8

]APPEND,9

]OUT,TEST

]
```

```
#*SORTER,I=34,C=2,T=2,P
COUNTER VALUES-
      0=          7
      1=          0
      2=          0
      3=          0
      4=          0
      5=          0
      6=          0
      7=          0
      8=          0
      9=          0
     11=          7
     12=          8
   REJECT=        0
   TOTAL=        22

#EDIT

]FIN 12

]APPEND,11

]APPEND,0

]OUT,TEST
```

(COLUMNS 2 AND 3 ARE NOW SORTED ALPHABETICALLY)

```
]
#*SORTER,I=34,C=1,T=1,P
```

(SAME PROCEDURE FOR COLUMN 1.)

```
COUNTER VALUES-
      0=          0
      1=          2
      2=          2
      3=          2
      4=          3
      5=          3
      6=          2
      7=          2
      8=          3
      9=          3
     11=          0
     12=          0
   REJECT=        0
   TOTAL=        22

#EDIT

]FIN 0
```

```
]APPEND,1

]APPEND,2

]APPEND,3

]APPEND,4

]APPEND,5

]APPEND,6


]APPEND,7

]APPEND,8

]APPEND,9

]OUT,TEST

]
#*SORTER,I=34,C=1,T=2,P
COUNTER VALUES-
        0=           8
        1=           0
        2=           0
        3=           0
        4=           0
        5=           0
        6=           0
        7=           0
        8=           0
        9=           0
       11=           5
       12=           9
   REJECT=           0
   TOTAL=           22

#EDIT

]FIN 12

]APPEND,11


]APPEND,0

]OUT,TEST
```

```
]LIST                                   (DATA IS NOW SORTED ALPHABETICALLY
     :ASD                                ON COLUMNS 1-3.)
     :BNM
     :CVB
     :DFG
     :ERT
     :FGH
     :GHJ
     :HJK
     :IOP
     :JKL
     :MQW
     :NMQ
     :QWE
     :RTY
     :SDF
     :TYU
     :UIO
     :VBN
     :WER
     :XCV
     :YUI
     :ZXC

]
#LOGOFF
 TIME 8.800 SECONDS MFBLKS 68 COST $0.95
```

EXAMPLE 3

```
7
9JOB,80000,ZIP,SAVE FOR KAY PORTER
8COPY,O=FORKAY
QWE
WER
ERT
RTY
TYU
YUI
UIO
IOP
ASD
SDF
DFG
FGH
GHJ
HJK
JKL
ZXC
XCV
CVB
VBN
BNM
NMQ
MQW
77
88
7LOGOFF
8
```

(LISTING OF DATA CARDS WHICH ARE ENTERED ON CARD READER.)

EXAMPLE 3

```
#EDIT

]FIN FORKAY

]LIST
     :QWE
     :WER
     :ERT
     :RTY
     :TYU
     :YUI
     :UIO
     :IOP
     :ASD
     :SDF
     :DFG
     :FGH
     :GHJ
     :HJK
     :JKL
     :ZXC
     :XCV
     :CVB
     :VBN
     :BNM
     :NMQ
     :MQW
```

FORKAY IS THE NAME OF DATA FILE CARDS THAT HAVE BEEN ENTERED FROM CARD READER. WE LIST THEM TO SEE OUR DATA LIST.

```
]
#EQUIP,33=FORKAY
#*SORTER,I=33,C=3,T=1,P
COUNTER VALUES-
      0=          0
      1=          1
      2=          2
      3=          3
      4=          3
      5=          3
      6=          3
      7=          2
      8=          3
      9=          2
     11=          0
     12=          0
  REJECT=         0
  TOTAL=         22
```

THE SORTING CONTINUES AS IT DID IN EXAMPLE 2.

```
#EDIT
]FIN 0

]APPEND,1

]APPEND,2

]APPEND,3

]APPEND,4

]APPEND,5

]APPEND,6

]APPEND,7

]APPEND,8

]APPEND,9

]OUT,TEST

]
#EQUIP,34=TEST
#*SORTER,I=34,C=3,T=2,P
COUNTER VALUES-
      0=           5
      1=           0
      2=           0
      3=           0
      4=           0
      5=           0
      6=           0
      7=           0
      8=           0
      9=           0
     11=           9
     12=           8
   REJECT=         0
   TOTAL=         22

#EDIT

]FIN 12

]APPEND,11

]APPEND,0

]OUT,TEST

]
```

```
#*SORTER,I=34,C=2,T=1,P
COUNTER VALUES-
       0=          0
       1=          1
       2=          3
       3=          2
       4=          3
       5=          3
       6=          3
       7=          2
       8=          3
       9=          2
      11=          0
      12=          0
    REJECT=        0
    TOTAL=        22

 #EDIT

 ]FIN 0


 ]APPEND,1

 ]APPEND,2

 ]APPEND,3

 ]APPEM
 ]APPEND,4

 ]APPEND,5

 ]APPEND,6

 ]APPEND,7

 ]APPEND,8

 ]APPEND,9

 ]OUT,TEST
```

```
#*SORTER,I=34,C=2,T=2,P
COUNTER VALUES-
      0=           7
      1=           0
      2=           0
      3=           0
      4=           0
      5=           0
      6=           0
      7=           0
      8=           0
      9=           0
     11=           7
     12=           8
   REJECT=         0
   TOTAL=         22

#EDIT

]FIN 12

]APPEND,11

]APPEND,0




]OUT,TEST



#*SORTER,I=34,C=1,T=1,P
COUNTER VALUES-
      0=           0
      1=           2
      2=           2
      3=           2
      4=           3
      5=           3
      6=           2
      7=           2
      8=           3
      9=           3
     11=           0
     12=           0
   REJECT=         0
   TOTAL=         22

#EDIT

]FIN 0
```

```
]APPEND,1

]APPEND,2

]APPEND,3

]APPEND,4

]APPEND,5

]APPEND,6

]APPEND,7

]APPEND,8

]APPEND,9

]OUT,TEST

]
#*SORTER,I=34,C=1,T=2,P
COUNTER VALUES-
      0=            8
      1=            0
      2=            0
      3=            0
      4=            0
      5=            0
      6=            0
      7=            0
      8=            0
      9=            0
     11=            5
     12=            9
   REJECT=          0
   TOTAL=          22

#EDIT

]FIN 12

]APPEND,11


]APPEND,0

]OUT,TEST
```

```
]LIST
     :ASD
     :BNM
     :CVB
     :DFG
     :ERT
     :FGH
     :GHJ
     :HJK
     :IOP
     :JKL
     :MQW
     :NMQ
     :QWE
     :RTY
     :SDF
     :TYU
     :UIO
     :VBN
     :WER
     :XCV
     :YUI
     :ZXC

]

#LOGOFF
 TIME 7.427 SECONDS MFBLKS 58 COST $0.82
```